TTT/RR##

Development of a Prototype NOAA Grid

05/25/05

Principal Investigator: Mark Govett
Phone: 303-497-6278
Email Address: Mark.W.Govett@noaa.gov

Other Investigator 1 (Dan Schaffer, Daniel.S.Schaffer@noaa.gov)
Other Investigator 2 (Brian Gross, Brian.Gross@noaa.gov)
Other Investigator 3 (Chris Moore, Christopher.Moore@noaa.gov)

Proposal Theme: NGI

Funding Received – FY 2004 $147,280

Performance

We have completed the primary deliverable; the construction of a prototype NOAA grid. The grid includes two small clusters of nodes at the Forecast Systems Laboratory (FSL), a large High Performance Computing System at FSL (Ijet), a cluster of nodes at the Geophysical Fluid Dynamics Laboratory (GFDL) and a single node at the Pacific Marine Environmental Laboratory (PMEL). Access to the grid is enabled by use of a prototype NOAA Certificate Authority (CA), the second deliverable for the project. This CA is based on Public Key Infrastructure (PKI) and constructed from the Globus SimpleCA package. The third deliverable was to port the Weather Research and Forecast (WRF), Regional Ocean Modeling System (ROMS) and GFDL Climate Model (CM2) to the various grid nodes. These models have been ported to Ijet and the GFDL clusters.

The final milestone, analysis of the feasibility of coupling WRF and ROMS across the prototype grid (between FSL and GFDL), was not completed. The compute nodes on the GFDL cluster do not have public internet addresses. Therefore messages cannot be sent from those nodes to other grid nodes. The proposal called for using MPICH-G2, a grid-enabled version of the Message Passing Interface (MPI) communications library to implement cross-grid communication. However, it turns out that MPICH-G2 requires complete connectivity between all the nodes. In lieu of this cross-grid coupled run, an attempt to coupled WRF and ROMS across the TeraGrid is underway.

Additional details on this effort as well as related papers are available at the grid project web-site.

Project Summary

As mentioned, the major accomplishment was construction of a prototype NOAA grid. Connectivity of grid nodes is enabled by firewall holes opened between the various grid nodes for designated ports. The ports correspond to services that allow remote job submission, remote login and remote file transfers across the grid. The grid is constructed using the Globus Toolkit and related third-party packages. Globus Toolkit (GT) 3.2.1 is currently in operation.

A rudimentary grid scheduler has been constructed using Globus and a Sun Grid Engine (SGE) job manager provided by the London e-Science Centre. The meta-scheduler allows remote jobs to be executed, including the staging in or out of any necessary files. However, it does not provide status information on the remote batch systems. Nor does it choose the optimal run location; the user must select the site by hand. Jobs are specified in a generalized XML format where the user requests a number of nodes, a queue, a maximum run-time, etc. These parameters are converted into system specific values that are passed on to the remote SGE batch system. The scheduler has been used to execute the WRF/ROMS and CM2 models on Ijet and the GFDL cluster.

Grid users can also execute simple commands to copy files between any of the grid nodes and execute remote grid login sessions.

The Certificate Authority providing grid security continues to be based on the Globus SimpleCA package. Now, however, grid CA certificates are stored in a MyProxy database located on one of the grid nodes. MyProxy provides services that enable users to download short-duration certificate proxies that enable grid access. An attempt was also made to implement a version of MyProxy that interfaces with one-time-password token cards. The idea is that the user would be required to enter a one-time-password in order to retrieve a certificate proxy. Although the MyProxy developers successfully demonstrated this capability on their own systems, it failed to work on the prototype grid. An alternative means for one-time-password access to MyProxy is discussed in the section entitled Future Direction.

A meta-scheduler enables grid users to query the status of grid compute resources and to submit jobs to the best available or specific platforms. Several grid meta-schedulers were investigated for use by NOAA grid users. Unfortunately, none of these efforts have come to fruition so far.

1. ClusterResources, Inc., (SILVER). This meta-scheduler works with Moab workload managers at the various grid sites. These managers, in turn, operate with Torque Portable Batch System (PBS), IBM Loadleveler, Platform LSF and other batch systems. However, support for SGE is not currently provided and it is unclear when it will be added. For demonstration purposes, Torque PBS and Moab were built on the small clusters on the grid. A Beta-version of the SILVER meta-scheduler was deployed on one of the grid clusters. It was possible to execute grid jobs using SILVER.

2. Edinburgh Parallel Computing Center, Job Scheduling Hierarchically (JOSH). This meta-scheduler is designed to operate with SGE batch systems only. It takes a client-server approach with servers located at each of the remote compute sites. Users can choose a particular grid compute site or allow JOSH to select an optimal one based on system and grid network loads. JOSH only worked on some of the grid nodes and file staging did not work properly.

3. Community Scheduler Framework (CSF). This is a meta-scheduling initiative funded by Platform Computing, Inc. The feasibility of CSF deployment was briefly examined. However it was soon discovered that CSF is not supported for Globus 3.2.1 due to the instability of this version of Globus. Moreover, the CSF developers rely upon community efforts to provide plugins for batch systems other than Platform's own LSF. There do not currently appear to be any organized efforts to provide these plugins.

4. University of Wisconsin, Condor-G. This meta-scheduler failed to work at all on the prototype grid. The Condor-G developers attribute this to the fact that we tested it using Globus 3.2.1.

The biggest impediment to greater success with this project was the lack of stability of the software tools used to construct the grid. At the GlobusWorld 2005 conference in February 2005, it finally became clear that the Globus developers were not providing the level of support for GT 3.2.1 needed by users despite the fact that it was advertised as a "stable release". Instead, the developers have been focusing their efforts on GT 4.X. This became problematic for the NOAA grid efforts directly and also indirectly during our attempts to use third-party software built on top of the Globus Toolkit. This lack of support contributed substantially to the failure to successfully implement a meta-scheduling capability. Even the rudimentary scheduler we developed is not completely reliable and is likely to be hampered by portability issues were the grid to be expanded. These problems might have been avoided if we had discussed the stability of Globus with the third-party software developers earlier on.

Success has also been hampered by various security issues at the NOAA sites. GFDL, FSL and PMEL each have different methods for securing firewalls and providing remote access. This heterogeneity makes it difficult to cleanly, securely provide the necessary firewall holes needed to construct the grid. Also, the security implementations for the HPCS's at FSL and GFDL have flaws. At both sites, secure token cards are required for remote access to the HPCS's but not for local access. Automatic disconnections of idle login sessions are not implemented at either site. At GFDL, the HPCS home directories are shared with individual user workstations. Finally, the security implementations are in flux. Just three days after two nodes located behind the PMEL firewall were successfully added to the grid, they became unavailable due to a change in the firewall implementation. These issues all impede development of the mutual trust between site administrators needed to implement the grid. Successful development of a NOAA grid will require uniform, stable security implementations by the participating sites.

Expenditure Summary

Figure 1. Expenditure report for TeraGrid project. This table is not complete!!!

Future Direction

Additional work remains to be done on this project. Firstly, indications are that GT 4.X is much more stable and better supported than GT 3.X. Therefore, one of the first tasks will be to re-implement the current grid with GT 4.X. This should be relatively easy. A GT4/SGE job manager is already available. GT 4.X implementations of CSF, Condor-G and JOSH are all under development. ClusterResources is also actively helping us to stabilize the SILVER meta-scheduler. Thus, all of these meta-scheduling solutions are candidates for a NOAA grid. The winner may partially depend on the batch system(s) chosen for the HPCS sites by the winning procurement vendor.

A second task is to expand the grid to include the three NOAA HPCS's. At the grid project web-site, there is a proposal we have developed for providing uniform, secure access to a NOAA grid. The key features are as follows:

1. A security perimeter will be established around the grid machines. Access to any of the nodes will be possible only via secure token cards. However, once on a grid node, a user with a grid certificate proxy will have access to other grid nodes without needing to enter a new one-time-password.

2. Grid site firewalls will be configured to only allow connectivity between nodes via a set of well-defined ports/services.

3. The current certificate authority will continue to be used. A single CA administrator responsible for verifying the legitimacy of prospective users will be identified. Grid certificates will be stored in MyProxy implementations on a few highly secure grid nodes. Users could download certificate proxies with the use of secure token cards. The actual certificates would never leave the secure MyProxy servers.

4. Each site administrator will be responsible for granting users access to resources as needed.

As the proposal discusses, the various site administrators would not be currently willing to institute the grid access described due to the security flaws discussed earlier. These security flaws would need to be corrected to make the proposal viable.

Another future task is to create a job priority scheme where some grid jobs could have higher priority than others. Allowing this requires making it possible to preempt models such as WRF, ROMS, and CM2. Since checkpoint restart is not available for these kinds of MPI jobs an alternative method is required. One idea to be investigated is modifying the model code so that it frequently checks if the job executing this code has been killed.

If so, then the model would save state and exit gracefully. The batch scheduler would have to be modified so that it informs a job that it has been killed with a few minutes of warning.

Completion of the coupled modeling over the TeraGrid is also planned. WRF will be run at the San Diego Supercomputing Center (SDSC) while ROMS is executed on a TeraGrid node at the National Center for Supercomputing Applications (NCSA). Cross-grid communications performance will be measured.

An FY '05 proposal has been submitted to HPCC to upgrade the grid to use Internet Protocol version 6 (IPv6). This work should simplify the means by which the various grid machines can address each other.

Finally, development of a java-based model-testing portal is underway as part of the Modeling and Assimilation Portability Project. This portal will enable users to configure and execute large numbers of runs of WRF and other models. Follow-on work to this grid effort will be to "grid-enable" the portal. By this we mean making it possible for users to, from the portal, access the grid via a single token card based sign-on and then execute test runs on any of the grid nodes. Web and grid capabilities will be added to enable discovery of data needed for model initialization, boundary conditions, verification, etc. We eventually envision a portal as a tool that could manage hundreds or thousands of runs needed to support model development and the evaluation or even simulation of observing systems.